REMARKS

Reconsideration of the application is respectfully requested.

Addressing first some informalities, the title of the invention has been changed as requested by the Examiner. In addition, a correction has been made to the specification regarding the reference number for decode stage 440. Finally, the abbreviation "IP" in claim 5 has been replaced with the intended full text.

Turning now to the art rejections, the claims stand rejected as being anticipated under 35 U.S.C. §102(e) by U. S. Patent 6,374,349 to McFarling ("McFarling '349"). Applicant respectfully disagrees with the rejection, particularly with respect to claim 3, because McFarling does not teach or suggest a method for branch prediction as recited in claim 3 in which a replacement field for matching entry of a local branch history table is updated only if a prediction made using a saturating counter branch predictor is incorrect, indicating that the entry is used to make a prediction, where the overall method for branch prediction involves the use of a combination of a saturating counter branch predictor and a local branch history table.

McFarling '349 describes a branch predictor with serially connected predictor stages. A selective cache entry **replacement** technique is described, at column 9 lines 45-55. However, this does not teach or suggest **updating a replacement field for a matching entry in a local branch history table only if a prediction using a saturating counter branch predictor is incorrect, indicating that the entry is used to make a prediction**. There is no teaching or suggestion to modify the branch prediction technique in McFarling so that replacement fields for matching entries in a local branch history table are updated as recited in claim 1.

As to dependent claim 10, McFarling '349 also does not teach or suggest a processor in which entry replacement logic is to update a replacement field for a matching entry in a local branch history table that provides a taken/not taken history in response to a hit, where the replacement field is updated only if the prediction from saturating counter branch prediction logic is incorrect.

Finally, dependent claim 17 is also not anticipated or obvious in view of McFarling '349 because there is no teaching or suggestion that means for updating a replacement field for a matching entry in a local branch history prediction table be provided, where the updating means is to update the replacement field only if a prediction from a means for providing a branch prediction that is based on the current state of a state machine is incorrect.

Applicant also respectfully disagrees with the rejection of dependent claim 4, where the method for branch prediction further involves fetching an instruction at a given address, and decoding the instruction so that at least one of the first and second predictions is available when the instruction is being decoded.

The Office Action, at page 9, states that McFarling does not explicitly teach this technique, although it would have been allegedly obvious modification, so as to meet the purpose of a branch prediction scheme which is to keep the pipeline full and free from stalls. For this motivation, a reference is made to "Computer Architecture A Quantitative Approach, by Patterson and Hennessy, 2d Ed., p. 167 ("Hennessy"), where a pipeline sequence is depicted for a predict-not-taken scheme. This, however, does not provide the needed motivation nor does it teach the capability recited in dependent claim 4.

The Examiner's attention is directed to the legend for Fig. 3.26 in Hennessy which illustrates the prediction scheme, and states, "when the branch is untaken, **determined during [instruction decode] ID**, we have fetched the fall-through and just continue. **If the branch is taken during ID**, we restart the fetch at the branch target. This causes all instructions following the branch to stall one clock cycle." [Emphasis added]. The point being made here is that in this particular pipeline, the branch is actually taken or not taken **at the instruction decode stage**. In contrast, Applicant has clarified that claim 4 is directed to the situation where if the instruction is a branch, a determination as to whether the branch is taken or not taken will not be available until the instruction has progressed beyond the decode stage. No new matter has been added, as this amendment is supported in the specification as filed, at page 12 lines 3-23. Accordingly, the situation described in Hennessy and cited by the Examiner is a clearly different scenario where the branch is either taken or not taken **at the decode stage** rather than in a later stage of the pipeline.

In addition, the particular scenario described in Hennessy is not similar to the techniques described in McFarling '349 because in Hennessy the prediction scheme is explicitly assumed to predict every branch as not taken, allowing the hardware to continue as if the branch were not executed. See Hennessy, second full paragraph as well as the third full paragraph. Accordingly, Applicant submits that Hennessy does not fairly teach or suggest to one of ordinary skill in the art faced with the mechanisms of McFarling '349 that the timing of the pipeline stages be modified to support the method recited in Applicant's claim 4.

In sum, a good faith attempt has been made to explain why the rejection in view of McFarling '349 is improper and to rewrite the claims so that all pending claims, namely claims 1, 2, 4-9, 11-16, 18, and 19 are submitted to be in condition for allowance.

Respectfully submitted,

BLAKELY SOKOLOFF TAYLOR & ZAFMAN LLP

Dated: April 24, 2003

Farzad E. Amini, Reg. No. 42,261

12400 Wilshire Boulevard
Seventh Floor
Los Angeles, California 90025
(310) 207-3800

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail with sufficient postage in an envelope addressed to: Assistant Commissioner for Patents, Washington, D.C. 20231 on April 24, 2003.

Jean Svoboda